6Д

УДК 621.865.8

О.В. Даринцев, А.Б. Мигранов
Институт механики УНЦ РАН, г. Уфа, Россия
ovd@imech.anrb.ru

Особенности реализации интеллектуальных алгоритмов планирования движений на бортовых вычислительных устройствах мобильных микророботов

В статье рассматривается использование нечетких интеллектуальных алгоритмов для решения проблемы планирования траекторий движения мобильного микроробота в условиях, когда вычислительные возможности бортовых электронных устройств ограничены размерами микроробота, а высокая степень изменчивости и неопределенности рабочей среды требует от системы управления интеллектуальных способностей при принятии решений.

Введение

В последние годы в работах многих авторов значительное внимание уделяется проблеме коллективного управления мобильными роботами и микророботами. Одной из перспективных областей массового применения мобильных микророботов является концепция «умной (интеллектуальной) пыли» – популярной в последнее время технологии миниатюрных автономных электронных устройств. Задачи управления автономными мобильными микророботами, особенно в условиях их массового применения, отличаются высокой степенью изменчивости и неопределенности окружающей обстановки и стохастическим характером процессов во внешней среде, что требует от системы управления интеллектуальных способностей при принятии решений. Поэтому для управления движением таких систем в последние годы широкое распространение получили методы решения, в частности, на основе нейронных сетей и генетических алгоритмов [1-4].

Реализация бортовых систем управления «большими» мобильными роботами на основе перечисленных подходов не вызывает особых технических затруднений, поскольку их размеры позволяют размещать на борту аппаратно-программные комплексы, ресурсы которых будут достаточными для реализации алгоритмов управления, работающих в реальном времени. В то же время на возможности бортовых вычислительных устройств микророботов накладываются более жесткие ограничения – размеры микророботов уже не позволяют размещать аналогичные по производительности аппаратно-программные комплексы. Таким образом, среди проблем массового применения микророботов в составе больших групп, насчитывающих сотни и даже тысячи микророботов, необходимо особо отметить сложность построения бортовой системы управления «отдельным» микророботом такой группы, так как методы и алгоритмы планирования и управления групповыми действиями микророботов должны изначально разрабатываться с учетом необходимости их реализации в реальном времени на основе имеющихся и ограниченных по мощности вычислительных ресурсов.

Целью данной работы является разработка алгоритмов управления для планирования движений и массового применения микророботов в составе больших групп на основе использования нетребовательных к аппаратным ресурсам интеллектуальных методов, обеспечивающих высокую скорость отклика в сочетании с такой же точностью, как у классических регуляторов. Отмеченным требованиям удовлетворяет использование принципов нечетких систем управления; сравнение результатов классического и нечеткого управления, проведенные во многих работах по применению теории нечетких множеств, показали, что нечеткое управление проще и эффективнее обычно применяемых классических методов [5]. К большому преимуществу нечеткого управления следует также отнести его ориентацию на цифровую реализацию.

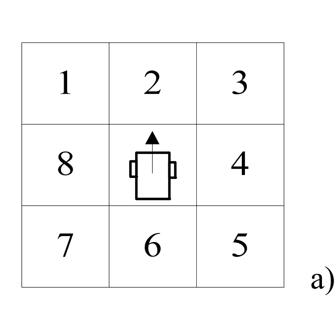
Проектирование базы нечетких лингвистических правил

При формировании базы нечетких лингвистических правил необходимо пройти следующие этапы проектирования: а) определить входы и выходы создаваемой системы; б) задать для каждой из входных и выходных переменных функции принадлежности; в) разработать нечеткие правила выводов. Рассмотрим подробнее каждый из этапов.

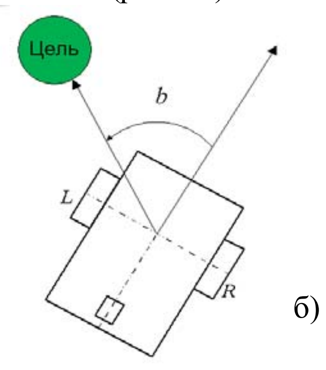
За входные сигналы нечёткого регулятора примем следующие параметры: **A** – свободные области рабочей зоны; *b* – угловое отклонение от цели.

Свободные области рабочей зоны **A** – это матрица, формируемая на основе поступающей информации от восьми датчиков, расположенных в каждом из 8-ми возможных направлений движения. На рис. 1 а) представлен порядок кодировки элементов матрицы. Элементы матрицы могут принимать значения «0» и «1», где «0» соответствует свободной области, а «1» соответствует препятствию в рабочей зоне.

Угловое отклонение от цели *b* – входная переменная, определяющая отклонение курса движения мобильного робота от целевого направления (рис. 1 б).



а)



б)

Рисунок 1 – а) кодировка элементов матрицы; б) угловое отклонение от цели

Выходным сигналами нечеткого регулятора являются:

V – линейная скорость мобильного робота;

ω – направление движения мобильного робота.

Направление движения мобильного робота ω – это выходная переменная, определяющая необходимое угловое смещение робота, при котором достигается достижение цели. Значение этого параметра необязательно должно совпадать со значением входной переменной *b*.

Входным и выходным сигналам соответствуют логико-лингвистические переменные, значения которых определяются термами-множествами (табл. 1).

Таблица 1 – Значения отдельных термов лингвистических переменных

Символическое обозначение	Значения термов-множеств	
NB	Negative Big	Отрицательное большое
NM	Negative Middle	Отрицательное среднее
NS	Negative Small	Отрицательное малое
Z	Zero	Нуль
PS	Positive Small	Положительное малое
PM	Positive Middle	Положительное среднее
PB	Positive Big	Положительное большое

Определим функции принадлежности для всех входных и выходных переменных системы.

Рассмотрим, как определяются функции принадлежности на примере входной *a* и выходной *V* переменных системы.

Входная переменная *a* (свободные зоны). Термы: СЗАДИ (NVB), СЗАДИ-СЛЕВА (NB), СЛЕВА (NM), ПРЯМО-СЛЕВА (NS), ПРЯМО (Z), ПРЯМО-СПРАВА (PS), СПРАВА (PM), СЗАДИ-СПРАВА (PB), СЗАДИ (PVB) (рис. 2).

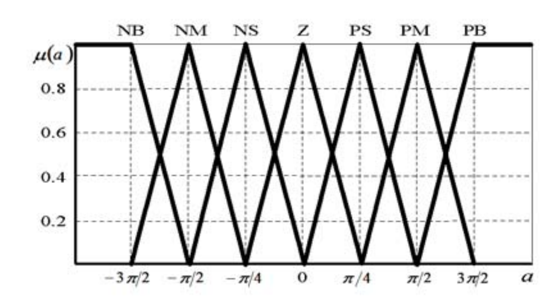


Рисунок 2 – График функции принадлежности лингвистической переменной *a*

Выходная переменная *V* (линейная скорость мобильного робота). Термы: нулевая (Z); малая (PS); средняя (PM); большая (PB) (рис. 3).

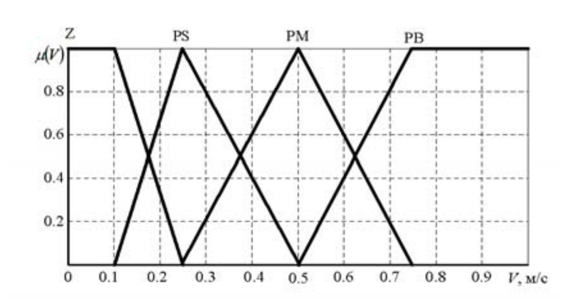


Рисунок 3 – График функции принадлежности лингвистической переменной *V*

На следующем этапе строится база знаний нечеткой системы, состоящая из продукционных правил и отражающая зависимость между входными и выходными термами-множествами. Для разрабатываемой нечеткой системы выбор нужного правила будет определяться угловым отклонением микроробота от цели и наличием свободных областей в рабочей зоне.

Всего в базе правил определено 72 правила – по девять вложенных правил для каждого из восьми значений углового отклонения робота от цели. Далее в качестве примера показана структура базы правил управления для ситуации, когда цель находится перед микророботом:

ЕСЛИ «Цель = Прямо» **ТО**

ЕСЛИ «Свободно = Прямо» **ТО** «Скорость = Малая» **И** «Направление = Прямо»
ИНАЧЕ ЕСЛИ «Свободно = Прямо-Слева» **ТО** «Скорость = Нулевая» **И** «Направление = Прямо-Влево»
ИНАЧЕ ЕСЛИ «Свободно = Прямо-Справа» **ТО** «Скорость = Нулевая» **И** «Направление = Прямо-Вправо»
ИНАЧЕ ЕСЛИ «Свободно = Слева» **ТО** «Скорость = Нулевая» **И** «Направление = Влево»
ИНАЧЕ ЕСЛИ «Свободно = Справа» **ТО** «Скорость = Нулевая» **И** «Направление = Вправо»
ИНАЧЕ ЕСЛИ «Свободно = Сзади-Слева» **ТО** «Скорость=Нулевая» **И** «Направление = Назад-Влево»
ИНАЧЕ ЕСЛИ «Свободно = Сзади-Справа» **ТО** «Скорость=Нулевая» **И** «Направление = Назад-Вправо»
ИНАЧЕ ЕСЛИ «Свободно = Сзади» **ТО** «Скорость = Нулевая» **И** «Направление = Назад»
ИНАЧЕ «Скорость = Нулевая» **И** «Направление = Прямо»

Как видно из этого примера, в базе правил в первую очередь выполняется поиск по переменной «цель», что позволяет эффективнее использовать вычислительные ресурсы бортовой микроЭВМ. Аналогично строятся базы нечетких правил управления для других возможных ситуаций расположения цели относительно робота (цель перед роботом и слева, цель перед роботом и справа, цель слева, цель справа и т.д.). В зависимости от конфигурации рабочего пространства также выделено несколько форм поведения робота: «движение к цели», «обход препятствия», «обход стены», «выход из тупика».

Планирование движений на основе форм поведения

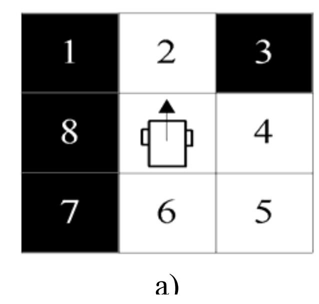
Предложенные правила нечетких знаний позволяют роботу однозначно выбирать наиболее эффективное направление движения для текущего положения робота. Однако могут возникнуть ситуации, при которых выбранное направление движения в конечном счете оказывается тупиковым. Правила в базе нечетких знаний не обеспечивают предотвращения таких ситуаций, поскольку они определяют решение для текущего местоположения робота, не учитывая конфигурацию всей рабочей области. Для решения этой проблемы предложено использовать планирование движений на основе форм поведения.

Под формой поведения понимается определенная последовательность действий, определяющих особенности движения робота в течение некоторого периода времени. Всего определено три формы поведения: «движение к цели», «обход стены», «выход из тупика». Поведение «движение к цели» используется при отсутствии препятствий на пути к целевой точке или при такой конфигурации рабочей зоны, которая не создает неоднозначных вариантов выбора эффективного направления движения. При обнаружении сплошного препятствия в виде «стены» выбирается вторая форма поведения – «обход стены». Если в процессе движения робот выходит за границы рабочей зоны, не достигнув желаемого положения, либо он отдаляется от целевой точки, то используется третья форма поведения – «выход из тупика».

Выполнение соответствующих алгоритмов выбранной формы поведения позволяет сформировать корректирующий курс, исключающий неоднозначные направления движения. Исключение таких направлений движения достигается временной блокировкой элементов в матрице свободных зон. Применение предложенных нечетких правил к сформированной таким образом матрице с отдельными заблокированными элементами позволяет роботу выбрать единственно верное направление движения.

Поясним сказанное на следующем примере. На рис. 4а) показана конфигурация рабочей области, в которой препятствия обнаружены в направлениях ПРЯМО-СЛЕВА, СЛЕВА, СЗАДИ-СЛЕВА, ПРЯМО-СПРАВА. Таким образом, робот может двигаться в любом из оставшихся направлений: ПРЯМО, НАПРАВО, НАЗАД-ВПРАВО, НАЗАД. Допустим, что робот должен двигаться только в направлении НАПРАВО. Для выполнения этого условия зоны 2, 5, 6 матрицы заблокируем (рис. 4б). Далее к сформированной таким образом матрице применяются знания базы нечетких правил, которые позволяют исключить неоднозначные направления движения.

Рассмотрим подробнее форму поведения «движение к цели». Как было уже отмечено, эта форма поведения используется при отсутствии препятствий или такой конфигурации рабочей зоны, которая не создает неоднозначных вариантов выбора эффективного направления движения. Пример такой конфигурации показан на рис. 5.



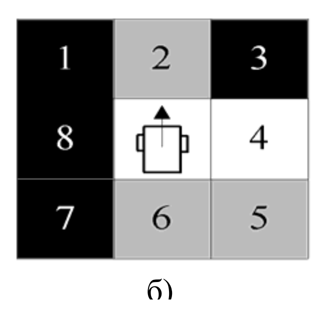


Рисунок 4 – Исключение неоднозначных направлений движения

В данной ситуации не возникает неоднозначностей в выборе направления движения. После проведения операции блокировки к зонам 1, 5 и 7 имеется три возможных направления движения: ПРЯМО-ВПРАВО, ПРЯМО и НАПРАВО. Рассмотрим правило в базе знаний, работающее в данной ситуации:

ЕСЛИ «Цель = Слева» ТО

ЕСЛИ «Свободно = Слева» ТО «Скорость = Нулевая» И «Направление = Влево»
ИНАЧЕ ЕСЛИ «Свободно = Прямо-Слева» ТО «Скорость = Нулевая» И «Направление = Прямо-Влево»
ИНАЧЕ ЕСЛИ «Свободно = Сзади-Слева» ТО «Скорость=Нулевая» И «Направление = Назад-Влево»
ИНАЧЕ ЕСЛИ «Свободно = Прямо» ТО «Скорость = Малая» И «Направление = Прямо»
ИНАЧЕ ЕСЛИ «Свободно = Сзади» ТО «Скорость = Нулевая» И «Направление = Назад»
ИНАЧЕ ЕСЛИ «Свободно = Прямо-Справа» ТО «Скорость = Нулевая» И «Направление = Прямо-Вправо»
ИНАЧЕ ЕСЛИ «Свободно = Сзади-Справа» ТО «Скорость=Нулевая» И «Направление = Назад-Вправо»
ИНАЧЕ ЕСЛИ «Свободно = Справа» ТО «Скорость = Нулевая» И «Направление = Вправо»
ИНАЧЕ «Скорость = Нулевая» И «Направление = Прямо»

Следовательно:

ЕСЛИ «Цель = Слева» ТО

ЕСЛИ «Свободно = Прямо» ТО «Скорость = Малая» И «Направление = Прямо»

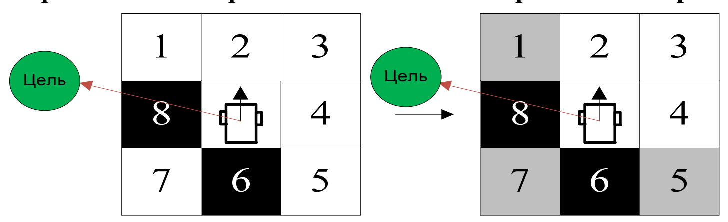


Рисунок 5 – Конфигурация рабочей зоны для формы поведения «движение к цели»

Таким образом, робот принимает решение двигаться в прямолинейном направлении. Алгоритм работы (рис. 6) основан на проверке наличия препятствий с правой (зоны ПРЯМО-СПРАВА, СПРАВА, СЗАДИ-СПРАВА) или с левой (зоны ПРЯМО-СЛЕВА, СЛЕВА, СЗАДИ-СЛЕВА) сторон робота. В случае обнаружения сплошного препятствия выбирается форма поведения «обход стены». В противном случае продолжается выполнение поведения «движение к цели».

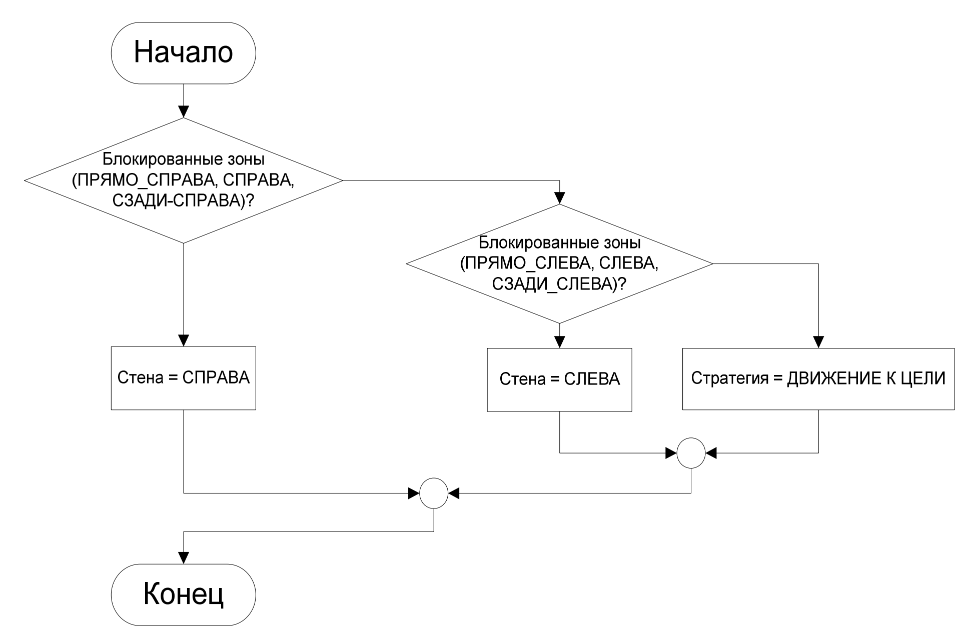


Рисунок 6 – Блок-схема алгоритма формы поведения «движение к цели»

Рассматриваемая форма поведения основана только на правилах в базе знаний и не формирует корректирующего курса движения. Комплекс правил в базе знаний стремится управлять роботом так, чтобы его угловое отклонение от цели было минимальным и большую часть времени робот двигался прямолинейно к цели. Поэтому при составлении базы правил было принято два условия:

– если угловое отклонение от цели больше нуля, то робот останавливается и изменяет свою ориентацию так, чтобы двигаться прямолинейно к цели;

– если угловое отклонение от цели равно нулю, то робот развивает максимальную возможную скорость и прямолинейно движется к цели.

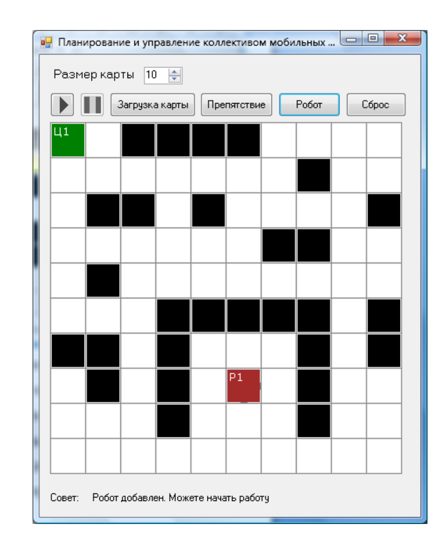
Данные условия обеспечивают по возможности наиболее прямолинейную форму движения и соответственно оптимальность по затрачиваемому времени и пройденному расстоянию.

Результаты моделирования

Некоторые из результатов моделирования разработанных алгоритмов планирования представлены на рис. 7. Рабочее пространство размерностью 10х10 представляет собой область со статическими препятствиями. Заданы исходное (Р1) и целевое (Ц1) положения робота.

Для движения к целевой точке первоначально используется форма поведения «движение к цели». На шаге 1.3, когда робот сталкивается с препятствием типа «стена», вступает в силу форма поведения «обход стены». Она выполняется до тех пор, пока робот не обнаружит конец стены в точке 1.10, начиная с которой вновь продолжается

выполнение формы поведения «движение к цели». Таким образом, последовательно изменяя различные формы поведения в зависимости от текущей ситуации, робот достигает целевой точки. Как видно из результатов, траектория движения робота также является в достаточной степени оптимальной по длине маршрута и затраченному времени на движение.



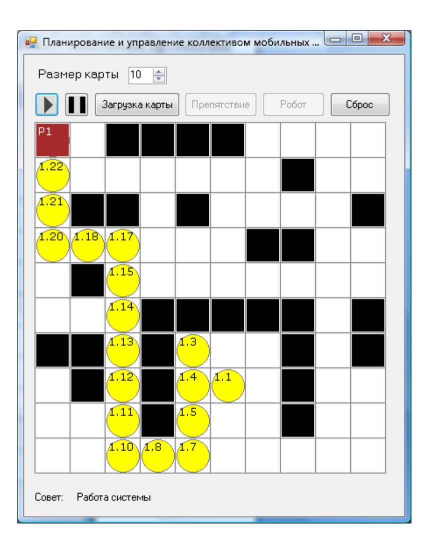


Рисунок 7 – Результаты моделирования

Выводы

Как показали результаты экспериментов, важной отличительной чертой разработанных алгоритмов планирования являются умеренные затраты машинного времени и оперативной памяти микроЭВМ. Поэтому предложенные алгоритмы могут быть использованы при построении бортовых систем управления микророботами. Вычислительные затраты практически не зависят от размерности матрицы конфигурации рабочей зоны, а оптимизация используемого объема памяти микроЭВМ для больших размерностей рабочей области может быть достигнута отображением в матрице локальной зоны робота.

Рассмотренный подход в будущем предполагается использовать как для управления отдельным мобильным микророботом, так и для управления большими группами микророботов. Планируется также провести сравнительный анализ предлагаемого подхода с другими интеллектуальными алгоритмами планирования и управления, реализованными на основе генетических алгоритмов и нейронных сетей.

Благодарность

Авторы выражают признательность за финансовую поддержку проводимых исследований ОЭММПУ РАН (Программа № 15 фундаментальных исследований).

Литература

1. Даринцев О.В. Система планирования движения группы мобильных микророботов на основе генетических алгоритмов / О.В. Даринцев, А.Б. Мигранов // Известия РАН. Теория и системы управления. – 2007. – № 3. – С. 163-173.
2. Даринцев О.В. Использование нейронной карты для планирования траектории мобильного робота / О.В. Даринцев, А.Б. Мигранов // Искусственный интеллект. – 2009. – № 3. – С. 300-307.
3. Ziemke T. Adaptive behavior in autonomous agents / T. Ziemke // Presence. – 2003. – 7(6). – P. 564-587.

4. Glasius R. Neural Network Dynamics for Path Planning and Obstacle Avoidance / R. Glasius, A. Komoda and S. Gielen // Neural Networks. – 1995. – № 8, 1. – P. 125-133.
5. Kickert W.Y.M. Fuzzy theories on decision-making / W.Y.M. Kickert // Martinus Nijhoff Social Sciences Division. – Netherlands, 1978. – 182 p.

Literatura

1. Darincev O.V. Izvestija RAN. Teorija i sistemy upravlenija. № 3. 2007. S. 163-173.
2. Darincev O.V. Iskusstvennyj intellekt № 3. 2009. S. 300-307.
3. Ziemke T. Presence. 2003. 7(6). P. 564-587.
4. Glasius R.Neural Networks. 8/1. 1995. P. 125-133.
5. Kickert W.Y.M. Fuzzy theories on decision-making. Netherlands: Martinus Nijhoff Social Sciences Division. 1978. 182 p.

О.В. Даринцев, А.Б. Мігранов

Особливості реалізації інтелектуальних алгоритмів планування рухів на бортових обчислювальних пристроях мобільних мікророботів

У статті розглядається використання нечітких інтелектуальних алгоритмів для вирішення проблеми планування траєкторій руху мобільного мікроробота в умовах, коли обчислювальні можливості бортових електронних пристроїв обмежені розмірами мікроробота, а висока ступінь мінливості і невизначеності робочого середовища вимагає від системи управління інтелектуальних здібностей при прийнятті рішень.

O.V. Darincev, A.B. Migranov

Features of the Implementation of Intelligent Algorithms for Motion Planning on-board Computing Devices Mobile Microrobots

The article discusses the use of fuzzy intelligent algorithms for solving the problem of planning trajectories of mobile microrobots at a time when computing power on-boardelectronic equipment limited by the size microrobots, and a high degree of variability and uncertainty of the working environment requires the management of intellectual abilities in decision making.

Статья поступила в редакцию 22.06.2011.